ISSS603-CUSTOMER ANALYTICS & APPLICATIONS

Project Report

SUBMITTED BY

CHRIS THNG

DIYA NARESH RAO

PANKHURI DWIVEDI

PRIYANKA SHARMA

LIAO YUNXIA

HU YUNXIA

Table of Contents

Abstract

The dataset selected for this project is transactional data of a UK based online retail store.

E-commerce is an industry that relies heavily on analytics to drive the business and sustain in the competitive market. With huge volume of transactional data generated, e-commerce industry taps on the power of analytics to devise their strategies for customer acquisition, development and retention, which motivated us to explore the potential of this industry.

The given dataset is a 12-month time-series data & provides 15k+ individual transactions of 4k+ customers on a product level granularity, featuring a Catalog of over 3.5k products, which gives us an opportunity to determine the customer purchasing patterns, mining of the products frequently purchased together and identify the customer profile.

We aim to apply the following customer analytics techniques to the dataset to come up with managerial insights:

- Exploratory Analysis – To identify buying patterns
- Customer Segmentation – Classifying customers based on transactions
- Market Basket and Product Recommendation – To Cross Sell/Up Sell products
- Strategy based on the analysis – Managerial Recommendation on the derived insights

Related Research Articles

Latent Factorization

Since our dataset is sparse & we do not have explicit product ratings, we have used Latent Factorization to provide personalized recommendations. To build & evaluate our recommendation model, we've used the approach indicated in the paper Collaborative Filtering for Implicit Feedback Datasets by Hu, Koren, and Volinsky.

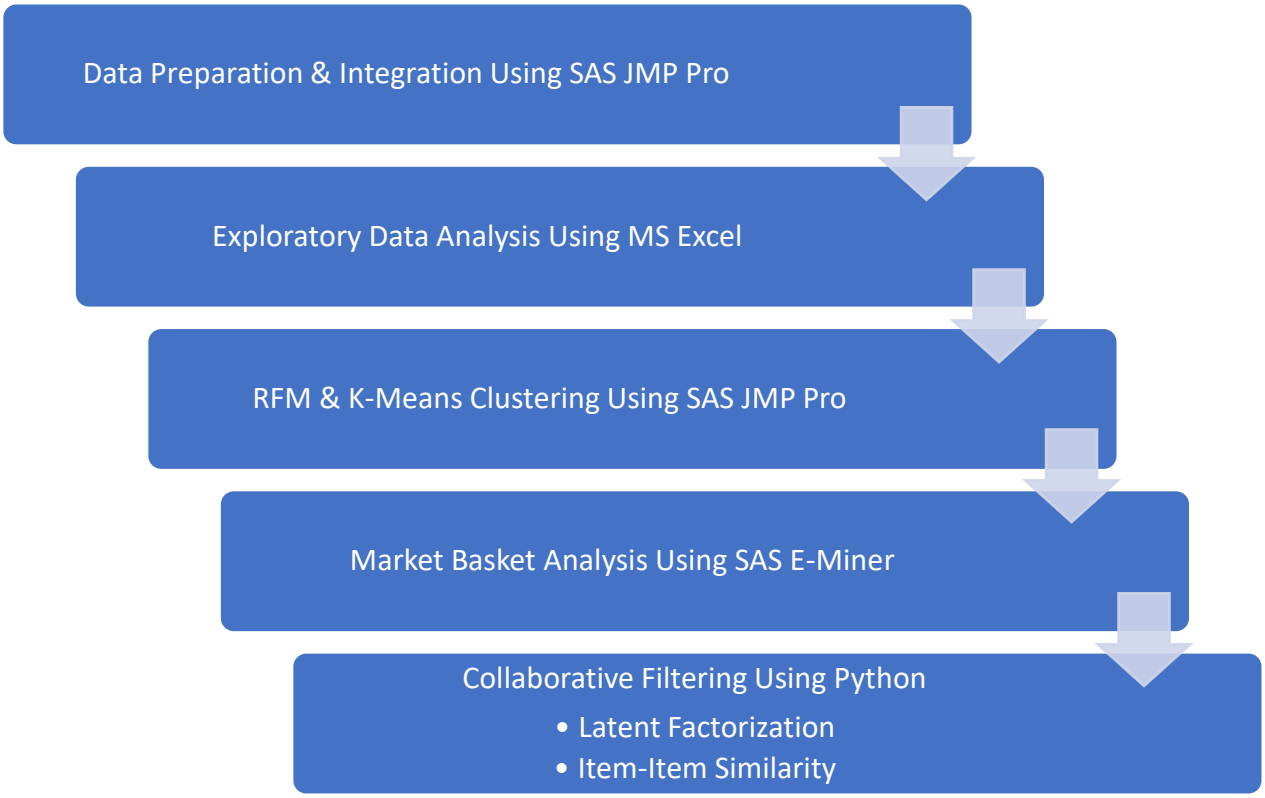
Item-Item Similarity

We have implemented item-item similarity to recommend products similar to the items selected by the user, using the below approach shared under "Session 7 Latent Factorization Recommender Systems" on e-learn:

Collaborative Filtering - Movie Ratings (Complex)

In order to evaluate the item-item similarity model, we have used the Graphlab package. We have referred to the blogpost by AARSHAY JAIN in a Quick Guide to Build a Recommendation Engine in Python

Visual Representation of Study Process



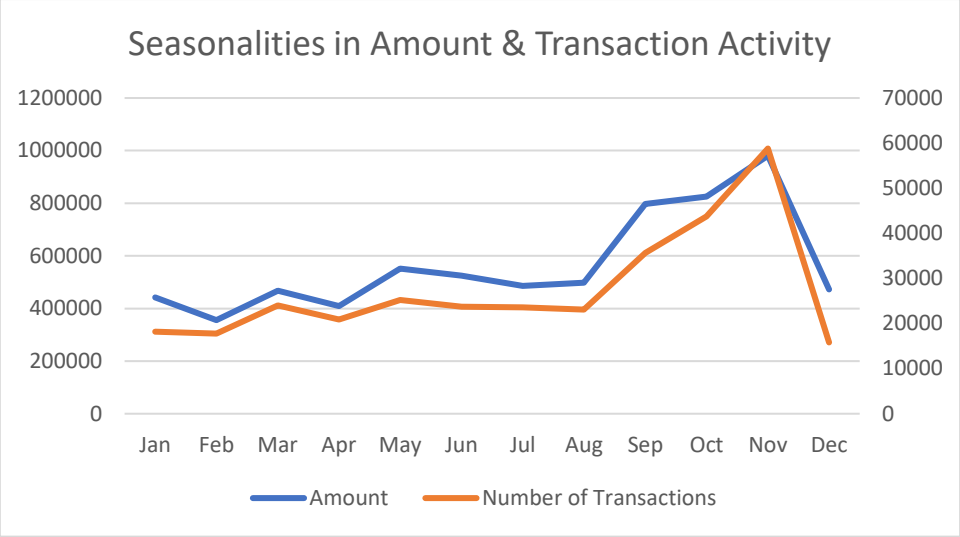
Data Preparation & Integration

Exploratory Data Analysis

The analysis is based on an online retail dataset that contains 354,321 records with 15,646 transactions made by 3,920 unique customers from Dec in Year 2010 to the end of 2011. Some explanatory data analysis was done before the data mining techniques were applied, so that general customer behaviours can be better understood.

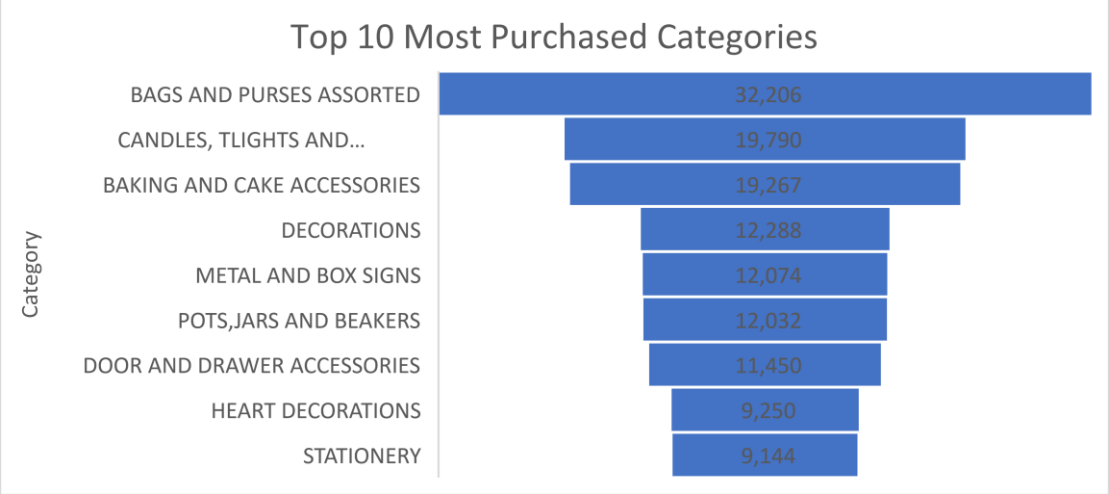
1. Univariate Analysis

1.1 Transactions and Amount



From data on hand, variable amount was generated based on unit price and quantity, counts of unique invoice number gave us the number of transactions. As it's shown in Figure, whether activity rate or consumptions are all affected by season. Obvious peaks were observed in Nov for both amount and number of transactions. It's probably caused by the fact that couple of traditional festivals fall on Nov and Dec, and consumers might be frequently purchasing gift items to prepare for the festivals.

1.2 Categories & Items



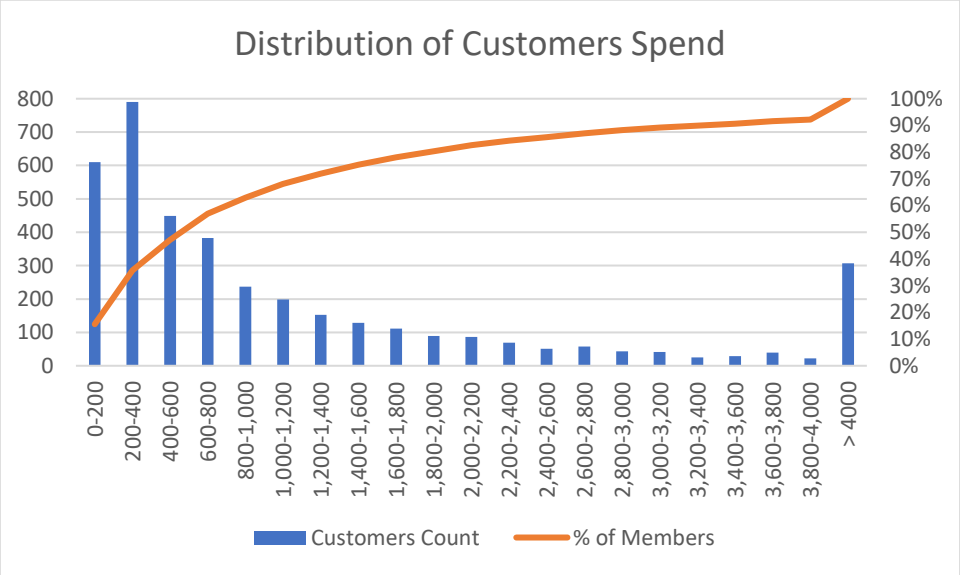
Top 10 most frequently purchased items are bags, candles and baking and decorations, which are very common as festive decorations. This has partially explained why peaks of sales and purchases took place in Nov.

For more accurate analysis on items people are buying, we've also looked at the indexes

Item	Quantity	Transactions	Amount	Quantity Per Transaction	Amount Per Transaction
MEDIUM CERAMIC TOP STORAGE JAR	76,919	176	80,291	437	456
WORLD WAR 2 GLIDERS ASSTD DESIGNS	49,182	425	12,137	115	28
JUMBO BAG RED RETROSPOT	41,981	1,464	77,371	28	52
WATER BOTTLE	40,713	6,731	17,3630	6	25
WHITE HANGING HEART T-LIGHT HOLDER	34,648	1,940	94,858	17	48

from top 10 items in purchasing quantity and purchasing amount. Table above has presented information about the top 10 items.

1.3 Customers



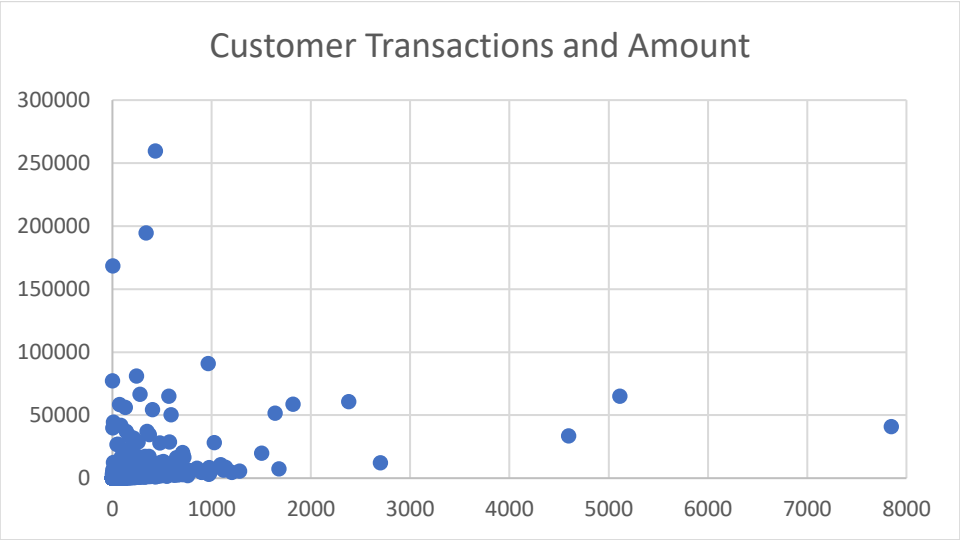
N	Min	1st Quartile	Median	3rd Quartile	Max	Mean	Std Dev
3,920	3,75	300.12	652.3	1,577.9	259,657.3	1,864.4	7,482.8

Descriptive Statistics

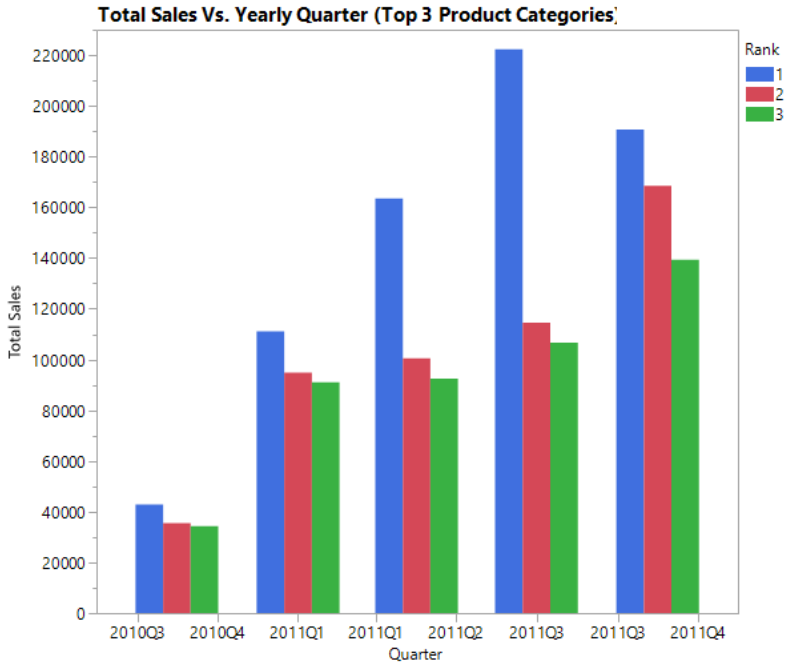
For purchasing amount among all the customers, as we can tell from Figure (), customers spend amount statistics are high right-skewed and there has been a polarization trend. It is

to say that most of customers may spend below 400, however, there is also a group of customers who spend for 4000 over a year. This also states the fact that a more strategic business framework needs be adopted for bettering targeting individuals.

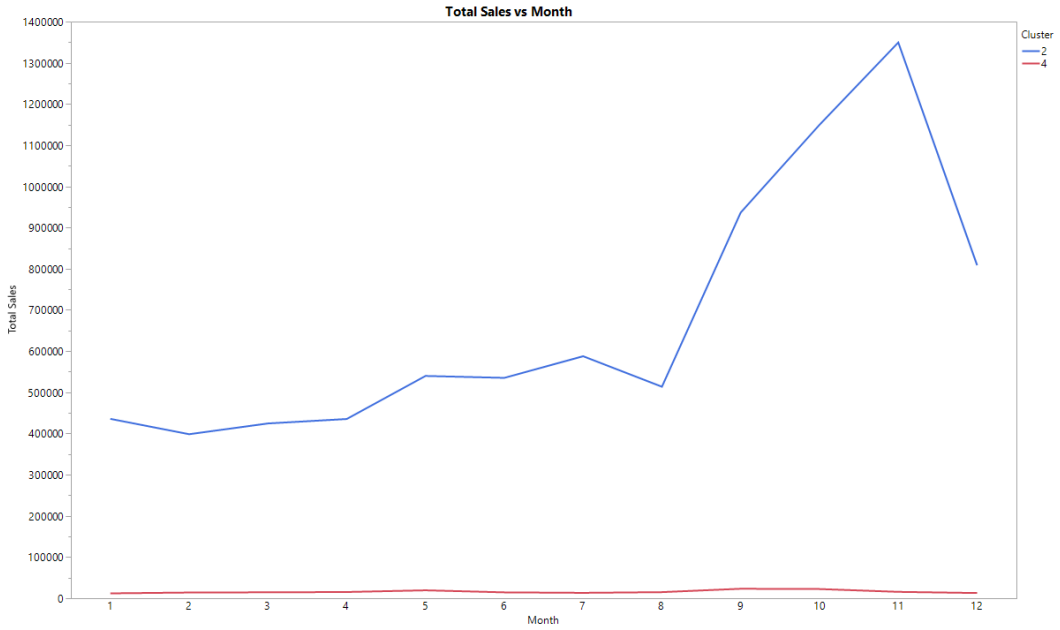
2. Bivariate Analysis



With aggregated customer data, we will be able to plot customer transactions and amount and track for correlations. From figure (), there is no apparent patterns for customer transactions and amount, no indication for higher activity rate will cause for higher amount in purchase. And thus, further segmentation on customers might still be required to achieve more detailed strategy on different customer groups.



The above graph shows the top 3 total sales per yearly quarter. It is observed that the Product Major Category 'BAGS AND PURSES ASSORTED' has maximum sales in 2011Q3.



The above graph depicts the difference in total sales between cluster 2 and cluster 4. Cluster 2 is our best performing cluster with the most valuable customers while cluster 4 contributes the least towards total sales, however contains 57% of our total customers constituting the mass market.

RFM Analysis

Data Preparation

The prepared data is utilized for RFM preparation. Figure below shows the data which was previously cleaned – erroneous data, non-UK data, categorized products.

onlineretail_productcat...
Source
Columns (15/0)
InvoiceNo
StockCode
Description
Cleaned column 2
Product Major Category
Quantity
InvoiceDate
UnitPrice
CustomerID
Country
Seasonality

11/0 Cols

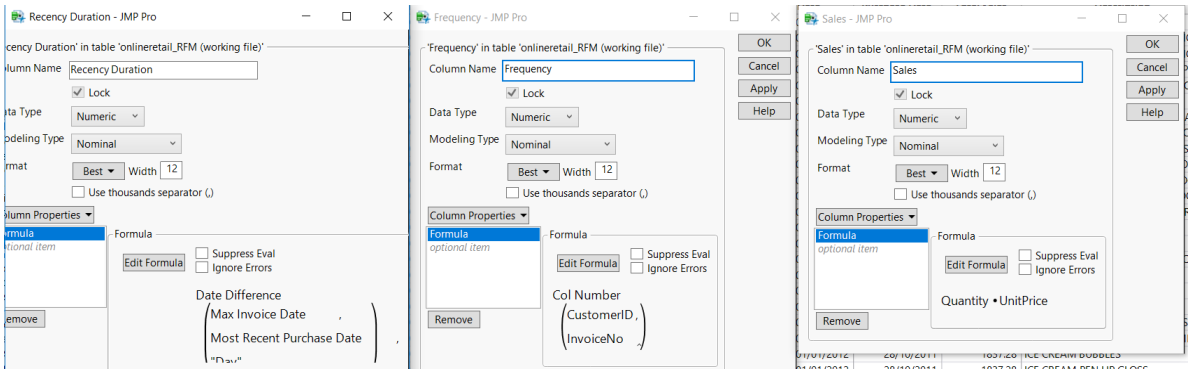
	InvoiceNo	StockCode	Description	Cleaned column 2	Product Major Category	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
1	564722	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	08/28/2011	8.95	16456	United King
2	564743	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	1	08/28/2011	8.95	14606	United King
3	564950	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	4	08/31/2011	8.95	17320	United King
4	564953	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	4	08/31/2011	8.95	17588	United King
5	565129	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/01/2011	8.95	15719	United King
6	565224	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/01/2011	8.95	14713	United King
7	565381	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/02/2011	8.95	16173	United King
8	565384	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	4	09/02/2011	8.95	13507	United King
9	565401	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	1	09/02/2011	8.95	16686	United King
10	565473	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/05/2011	8.95	16717	United King
11	565488	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/05/2011	8.95	15307	United King
12	565513	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	12	09/05/2011	8.15	14401	United King
13	565750	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/06/2011	8.95	14208	United King
14	565850	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/07/2011	8.95	14530	United King
15	566019	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	2	09/08/2011	8.95	17243	United King
16	566061	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	4	09/08/2011	8.95	13267	United King
17	566227	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	12	09/11/2011	8.15	14258	United King
18	566274	82486	3 DRAWER ANTIQUE WHITE WOOD CAB...	3 DRAWER ANTIQUE WHITE WOOD CABINET	CABINETS	1	09/11/2011	8.95	13884	United King

In order to prepare the data for RFM. The 3 key attributes of Recency, Frequency, Monetary must be extracted from the data. Based on Figure below, we run the following formulas in order to get the required data for each attribute.

(1) Recency: Max Invoice Date (End Date of Dataset) – Most Recent Purchase Date of Transaction per CustomerID

(2) Frequency: A count of number of Transaction IDs by CustomerID

(3) Monetary: (UnitPrice*Quantity) by CustomerID

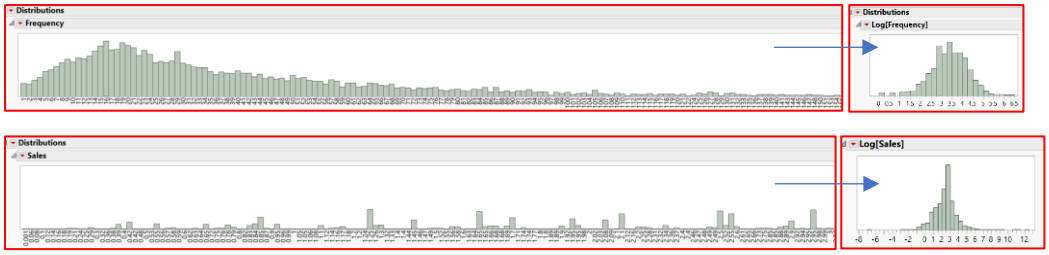


The formula was applied to the whole dataset. The sample of the resultant data is shown in figure below.

27/1 Cols	InvoiceNo	CustomerID	InvoiceDate	Quantity	UnitPrice	Recency Duration	Frequency	Sales
1	554065	18287	22/05/2011	30	1.25	-65	29	37.5
2	554065	18287	22/05/2011	24	0.42	-65	29	10.08
3	570715	18287	12/10/2011	24	0.42	-65	38	10.08
4	570715	18287	12/10/2011	24	0.42	-65	38	10.08
5	554065	18287	22/05/2011	6	2.55	-65	29	15.3
6	570715	18287	12/10/2011	36	0.42	-65	38	15.12
7	554065	18287	22/05/2011	12	1.45	-65	29	17.4
8	554065	18287	22/05/2011	12	2.55	-65	29	30.6
9	570715	18287	12/10/2011	6	2.1	-65	38	12.6
10	570715	18287	12/10/2011	48	0.39	-65	38	18.72
11	570715	18287	12/10/2011	24	0.85	-65	38	20.4
12	570715	18287	12/10/2011	24	0.85	-65	38	20.4
13	570715	18287	12/10/2011	48	0.83	-65	38	39.84
14	570715	18287	12/10/2011	20	1.25	-65	38	25
15	570715	18287	12/10/2011	4	3.75	-65	38	15

The purpose of RFM attributes was to give our dataset further robustness– in its raw form there were too few columns and also no customer behavioural data.

Upon further analysis of the results it was shown that the distribution amongst the Frequency and Sales values generated was not evenly distributed. Hence, Log transformation was applied in order assist K-Means clustering in the later stages.

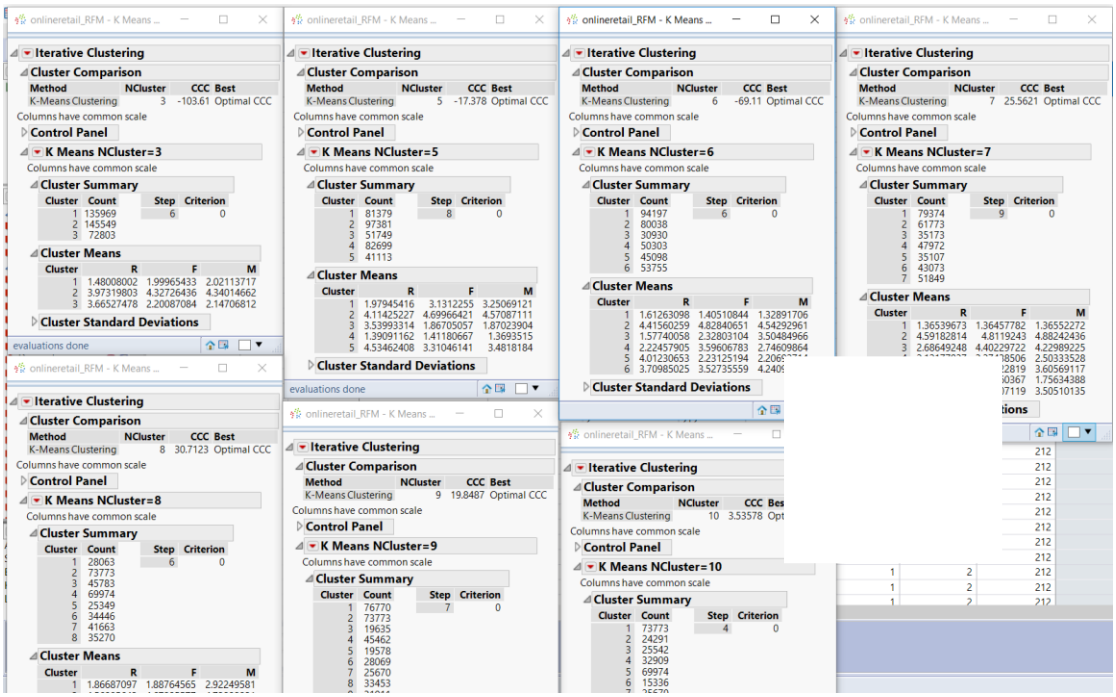


Cluster Analysis

The methodology is based on the Recency Frequency Monetary, RFM Model. With the data prepared – Recency, Frequency, Monetary values generated and normalized (F, M), K-Means is then applied to the RFM attributes in order to cluster customers based on their behavioural patterns – RFM.

K-Means

K-Means is applied on the R, F, M attributes obtained previously from the RFM data preparation step. We re-iterate from the number of clusters from 3 through to 10 based off the methodology of using CCC to determine the Optimal N. Once we identify a drop in CCC, we end this loop and identify the highest CCC – Optimal N.



Data is finalized into a summary table to be used for further analysis in the next part of the report – Exploratory Analysis (Overview) and Cluster Characterization.

Overview

Figure below provides an overview of the customers segmented by their respective clusters which is further broken down into R – Recency, F – Frequency, M – Monetary. Based on the figure, the clusters can be easily identified and characterized.

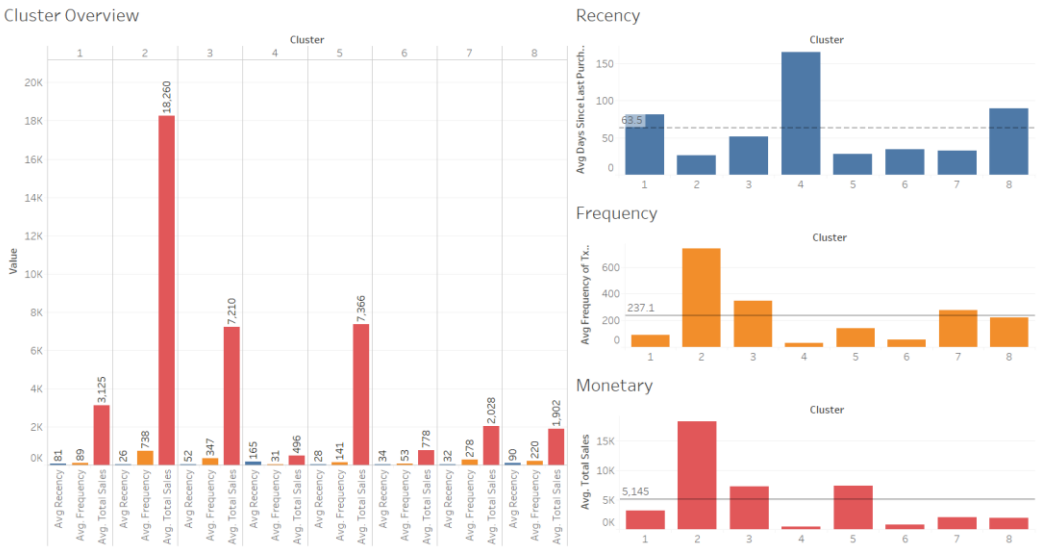


Figure below provides a breakdown of the number of customers represented per cluster. Other than Cluster 4, customers are fairly distributed within the remaining clusters.

Total # of Customers		
Cluster	# Customers	%
1	315	8%
2	100	3%
3	132	3%
4	2234	57%
5	180	5%
6	649	17%
7	150	4%
8	160	4%
Total	3920	100%

Figure below provides the respective averages for R, F, M in numerics and percentages. Based off this figure, clusters that are high performing are identified by increase in

percentages and vice-versa. Using this data, the cluster can then be characterized in a fair manner using data as the basis – done in a later stage.

	Average			RFM		
Cluster	Days	#	$	R	F	M
1	80.95	89.09	3124.55	29%	-1%	68%
2	26.45	737.73	18260.46	77%	716%	879%
3	51.76	346.84	7209.73	55%	284%	287%
4	164.87	31.32	495.71	-44%	-65%	-73%
5	27.80	140.83	7365.83	76%	56%	295%
6	34.31	53.08	777.59	70%	-41%	-58%
7	32.48	277.75	2028.03	72%	207%	9%
8	89.69	220.44	1901.90	22%	144%	2%
Average	114.74	90.39	1864.39			

	Total			Rank
Cluster	Days	#	$	($$$)
2		73773	1826046	1
5		25349	1325850	2
4		69974	1107414	3
1		28063	984234	4
3		45783	951684	5
6		34446	504656	6
8		35270	304303	7
7		41663	304205	8

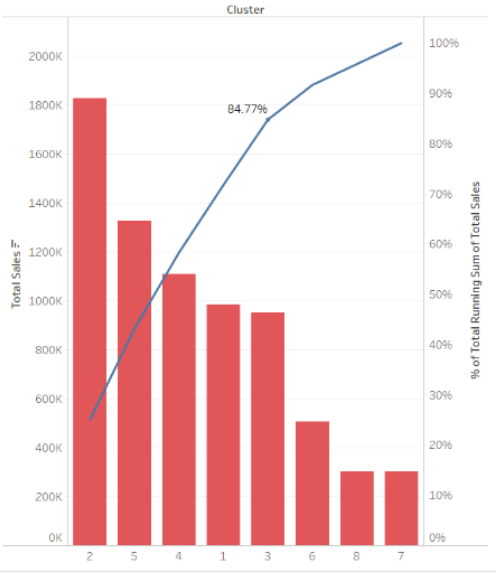


Figure above provides the total number of transactions and total sales per cluster. The clusters are then ranked based on the sales generated. Based off this data, we are able to determine the most valuable group in terms of sales.

Cluster Characterization

CLUSTER #	DESCRIPTION	OBJECTIVES
CLUSTER 1 NEED ATTENTION	Customers are average in all aspects, total sales, total transactions, average % of RFM.	Nurture, consider promotional campaigns
CLUSTER 2 CHAMPIONS	Most valuable customers, highest % in all aspects of RFM. Highest number of transactions as well as biggest contributor to total sales. Likely to be big corporations ordering high quality expensive items on a frequent basis.	Maintain and retain, consider cross-selling other high value products and making customers feel exclusive by

		providing access to preview sales, loyalty points, attractive seasonal bundles
CLUSTER 3 LOYALIST	Display similar purchasing behaviour to Cluster 2. However total sales pale in comparison at a rank of 5. They display high potential for growth.	Nurture, consider promotional campaigns as well as upselling
CLUSTER 4 MASS MARKET	This cluster has the lowest average % in terms of RFM. However, are the 3rd largest contributor to sales and highest % of customers in this group – 57%. Customers could likely be the mass market.	Lure by incentivising purchases 'We Miss You' discounts Feedback Mechanism to identify areas of improvement
CLUSTER 5 EXTRAVAGANT	Display similar purchasing behaviour to Cluster 3. However, are the 2nd largest contributor to total sales and lowest number of transactions. Likely to be corporations buying high quality expensive items, although less frequently. Fair potential for growth.	Retain and grow, consider relationship building campaigns to increase network of similar HNW companies and bulk buy promotions to increase frequency and also cross-selling to expose to other products
CLUSTER 6 NEW CUSTOMERS	Display similar purchasing behaviour as Cluster 4. However, are more recent customers. Likely to be customers living within the area of the shop. Total sales are average at a rank of 6 and make up the second highest # of customers at 17% of total.	Improve relations with personalised emails/offers Provide good recommendations, attracting them to buy more
CLUSTER 7 POTENTIALS	This cluster is ranked the last in terms of sales contribution. Small fraction of total customer base at 4%. Likely to be loyal customers, constantly frequenting the shop to buy inexpensive items. Interest lies in the cheaper products.	Nurture, consider upselling products and running promotional campaigns
CLUSTER 8 DORMANT	Display similar purchasing behaviour as Cluster 4. However, less frequent, less recent, even cheaper items. Likely to be loyal customers but belong to a slightly poorer class as compared to Cluster 7.	Send personalized emails to re-connect Recommend popular products/ renewals at discount

Market Basket Analysis

In this area of analysis, we have to first determine which segments to focus our resources on.

Based on our customer segmentation analysis:

1. **Cluster 4**: Mass market 59% of total customers,

A key segment due to majority of customer base belonging to this cluster, identifying strategic associations between products could potentially boost the negative R, F, M scores attached to this cluster

2. **Cluster 2**: Most valuable customers - top sales and transactions,

A key segment due to highest scores in R, F, M and identifying strategic associations between products could boost sales through cross-selling/up-selling

3. **Cluster 5**: High net-worth / "extravagant" customers - second highest sales,

A key segment due to high scores in R, M, lower frequency but bigger ticket item purchases and identifying strategic associations between products could boost sales through cross-selling

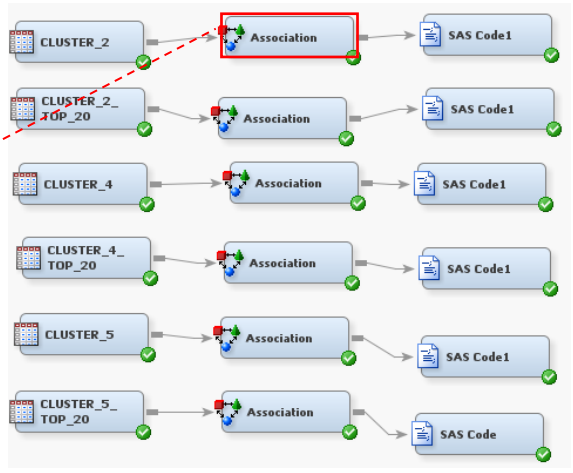
These 3 clusters have been identified for Production Association Analysis as they are our top 3 clusters with the most potential, capture a large customer base (63% of total) and contribute the a significant roughly 50% in revenue (top 3 in sales).

Data Preparation:

Target: Product Major Category
(this data column was prepared in the initial stage of data preparation in order to reduce the category levels from 3000 to 300)
ID: InvoiceNo

These 2 fields were used for association analysis conducted within SAS Enterprise Miner.

Property	Value
Association	
Maximum Items	4
Minimum Confidence Level	20
Support Type	Percent
Support Count	.
Support Percentage	10.0
Sequence	
Chain Count	3
Consolidate Time	0.0
Maximum Transaction Duration	0.0
Support Type	Percent
Support Count	.
Support Percentage	2.0
Rules	
Number to Keep	200
Sort Criterion	Default
Number to Transpose	200
Export Rule by ID	Yes
Recommendation	No



Additionally, in order to see the difference between a cluster which displayed an average of 200 product category levels per cluster, we took the top 20 most frequent product categories per cluster.

This will serve as a contrast to see the difference in association levels when conducted in this manner: Cluster(#N) (all levels) against Cluster(#N) (top 20 levels).

For each cluster type, we can see that by reducing the number of levels per category the result difference can be seen distinctly. The difference seen is a lower confidence and lift level. Hence, we would be not be excluding any levels from each cluster as results would be less optimal as compared to using the original dataset.

Cluster Analysis: Cluster 4
ALL: Selected

Obs	EXP_CONF	CONF	SUPPORT	LIFT	RULE	index	CHISQ	PVALUE
1	39.67	88.24	4.71	2.22	DOILIES ==> BAKING AND CAKE ACCESSORIES	38	230.128	0
2	24.41	77.92	4.52	3.19	STAR DECORATION ==> HEART DECORATIONS	1	395.223	0

Top 20

Obs	EXP_CONF	CONF	SUPPORT	LIFT	RULE	index	CHISQ	PVALUE
1	47.00	77.54	4.73	1.65	PHOTO FRAMES AND ACCESSORIES & DECORATIONS ==> CANDLES, TLIGHTS AND ACCESSORIES	71	94.101	0
2	47.00	76.34	5.17	1.62	HEART DECORATIONS & GARDENING ==> CANDLES, TLIGHTS AND ACCESSORIES	86	97.071	0

Based off the ALL Cluster Selection for Cluster 4,

The top 2 rules with highest confidence:

1. Doilies -> Baking and Cake Accessories (88.24%)
2. Star Decorations -> Heart Decorations (77.92%)

	Average			RFM		
Cluster	Days	#	$	R	F	M
1	80.95	89.09	3124.55	29%	-1%	68%
2	26.45	737.73	18260.46	77%	716%	879%
3	51.76	346.84	7209.73	55%	284%	287%
4	164.87	31.32	495.71	-44%	-65%	-73%
5	27.80	140.83	7365.83	76%	56%	295%
6	34.31	53.08	777.59	70%	-41%	-58%
7	32.48	277.75	2028.03	72%	207%	9%
8	89.69	220.44	1901.90	22%	144%	2%
Average	114.74	90.39	1864.39			

To target this Cluster 4: Mass Market, promotions can be made around LHS & RHS.

- Baking and Cake Accessories / Heart Decorations, more expensive items within this category can be bundled in order to up-sell

- Baking and Cake Accessories / Heart Decorations can be put on sale a different timing from Doilies / Star Decorations in order to drive all year-round sales / higher frequency purchasing patterns
- Seasonal promotions (e.g. Christmas, New Years, Halloween) will encourage them to spend more during such periods increasing Recency and overall sales
- Identify negative attributes per RFM, through feedback channels and improve upon such areas

Cluster Analysis: Cluster 2
ALL: Selected

Obs	EXP_CONF	CONF	SUPPORT	LIFT	RULE	index	CHISQ	PVALUE
1	27.47	82.89	5.49	3.02	STAR DECORATION ==> DECORATIONS	35	308.696	0
2	18.48	61.22	5.70	3.31	STATIONERY & DECORATIONS ==> NOTEBOOKS AND POCKETBOOKS	4	351.652	0

Top 20

Obs	EXP_CONF	CONF	SUPPORT	LIFT	RULE	index	CHISQ	PVALUE
1	24.62	62.50	5.56	2.54	POTS,JARS AND BEAKERS & DECORATIONS & BAKING AND CAKE ACCESSORIES ==> PANS, TINS AND KITCHEN ACCESSORIES	194	203.688	0
2	24.62	62.31	6.01	2.53	NOTEBOOKS AND POCKETBOOKS & CANDLES, TLIGHTS AND ACCESSORIES & BAKING AND CAKE ACCESSORIES ==> PANS, TINS AND KITCHEN ACCESSORIES	200	220.219	0

Based off the ALL Cluster Selection for Cluster 2,

The top 2 rules with highest confidence:

1. Star Decorations -> Decorations (82.89%)
2. Stationary & Decorations -> Heart Decorations (77.92%)

	Average			RFM		
Cluster	Days	#	$	R	F	M
1	80.95	89.09	3124.55	29%	-1%	68%
2	26.45	737.73	18260.46	77%	716%	879%
3	51.76	346.84	7209.73	55%	284%	287%
4	164.87	31.32	495.71	-44%	-65%	-73%
5	27.80	140.83	7365.83	76%	56%	295%
6	34.31	53.08	777.59	70%	-41%	-58%
7	32.48	277.75	2028.03	72%	207%	9%
8	89.69	220.44	1901.90	22%	144%	2%
Average	114.74	90.39	1864.39			

To target this Cluster 2: Most valuable customers, promotions can be made around LHS & RHS.

- Decorations / Heart Decorations, more expensive items in this category can be bundled in order to up-sell
- Decorations / Heart Decorations can be put on sale a different timing from Star Decorations / Stationary & Decorations in order to drive all year round sales / higher frequency purchasing patterns

- Consider including promotions such as buy 5 get 1 free in order to further drive frequency and sales in order to reward customers to further buy in bulk and try out other products at the same time
- Loyalty points of a higher tier can be assigned to these customers (e.g. Grab Platinum vs Grab Gold), and they can receive more benefits such as more discounts, priority delivery during promotional events
- Special benefits for belonging to this segment – exclusive to segment only (e.g. custom product engraving) these services are inexpensive but yet make the customer feel special. This will result in generation of loyalty and help with retention

Cluster Analysis: Cluster 5
ALL: Selected

Obs	EXP_CONF	CONF	SUPPORT	LIFT	RULE	index	CHISQ	PVALUE
1	32.24	63.21	6.09	1.96	KITCHEN BOXES, BINS AND CONTAINERS ==> BAKING AND CAKE ACCESSORIES	13	102.977	0.00000
2	32.24	60.42	6.59	1.87	PANS, TINS AND KITCHEN ACCESSORIES ==> BAKING AND CAKE ACCESSORIES	17	97.890	0.00000

Top 20

Obs	EXP_CONF	CONF	SUPPORT	LIFT	RULE	index	CHISQ	PVALUE
1	35.61	60.42	7.28	1.70	PANS, TINS AND KITCHEN ACCESSORIES ==> BAKING AND CAKE ACCESSORIES	11	73.237	0.00000
2	38.32	56.93	3.92	1.49	HEART DECORATIONS & DECORATIONS ==> CANDLES, TLIGHTS AND ACCESSORIES	38	21.562	0.00000

Based off the ALL Cluster Selection for Cluster 5,

The top 2 rules with highest confidence:

3. Kitchen Boxes, Bins and Containers -> Baking and Cake Accessories (63.61%)
4. Pans, Tins and Kitchen Accessories -> Baking and Cake Accessories (60.42%)

	Average			RFM		
Cluster	Days	#	$	R	F	M
1	80.95	89.09	3124.55	29%	-1%	68%
2	26.45	737.73	18260.46	77%	716%	879%
3	51.76	346.84	7209.73	55%	284%	287%
4	164.87	31.32	495.71	-44%	-65%	-73%
5	27.80	140.83	7365.83	76%	56%	295%
6	34.31	53.08	777.59	70%	-41%	-58%
7	32.48	277.75	2028.03	72%	207%	9%
8	89.69	220.44	1901.90	22%	144%	2%
Average	114.74	90.39	1864.39			

To target this Cluster 5: High net-worth / “extravagant” customers, promotions can be made around LHS & RHS.

- Baking and Cake Accessories, on a seasonality basis can be introduced to customers (e.g. Christmas Baking accessories – gingerbread man, Halloween Baking accessories). This will help to drive the frequency of such customers coming – increasing retention rate

- Loyalty points of a higher tier can be assigned to these customers (e.g. Grab Platinum vs Grab Gold), and they can receive more benefits such as more discounts, priority delivery during promotional events
- Consider including promotions such as buy 5 get 1 free in order to further drive frequency and sales in order to reward customers to further buy in bulk and try out other products at the same time
- Consider using recommendation engines such as collaborative filtering, association analysis in order to recommend related lower valued items that customers can consider, this cross-sell strategy will expose the customer to more products and potentially grow a new product category

Recommendation Using Collaborative Filtering

We have implemented 2 kinds of recommendation engines for the UI.

1. Latent Factorisation:

This recommendation engine follows the steps outlined in the paper by by Hu, Koren, and Volinsky.

In this approach we reconstruct the user-item matrix for the predicted ratings and then evaluate it using the AUC (Area under the curve) metric.

This recommender returns the top 10 items most likely preferred by the user which is provided as an input.

2. Item-Item Similarity:

In this approach we recommend the top items most likely preferred by the user by considering the user-item combination provided as input.

The predicted ratings are calculated by taking the weighted sum of the ratings of the top N items similar to the selected item and dividing by the sum of similarities.

Latent Factorization

Since our dataset is sparse & we do not have explicit product ratings, we have used Latent Factorization to provide personalized recommendations.

We will start with loading the data, which is saved in a csv file:

```
In [3]: item_lookup = pd.read_csv('item_lookup.csv')
```

```
In [4]: item_lookup.head()
```

Out[4]:

	StockCode	Description
0	10002	INFLATABLE POLITICAL GLOBE
1	10080	GROOVY CACTUS INFLATABLE
2	10120	DOGGY RUBBER
3	10123C	HEARTS WRAPPING TAPE
4	10123G	NaN

We've prepared a separate dataset of the quantity of individual items purchased by each customer in a csv file:

```
In [5]: grouped_purchased = pd.read_csv('grouped_purchased.csv')
```

```
In [7]: grouped_purchased.head()
```

Out[7]:

	CustomerID	StockCode	Quantity
0	15838	84347	1
1	18133	M	1
2	15749	47566B	1
3	16422	D	1
4	12908	M	1

From the above dataset, we've calculated the number of unique customers, items & the total quantity of purchases & created a sparse matrix as shown in the figure below:

```
customers = list(np.sort(grouped_purchased.CustomerID.unique())) # Get our unique customers
products = list(grouped_purchased.StockCode.unique()) # Get our unique products that were purchased
quantity = list(grouped_purchased.Quantity) # All of our purchases

rows = grouped_purchased.CustomerID.astype('category', categories = customers).cat.codes
# Get the associated row indices
cols = grouped_purchased.StockCode.astype('category', categories = products).cat.codes
# Get the associated column indices
purchases_sparse = sparse.csr_matrix((quantity, (rows, cols)), shape=(len(customers), len(products)))
```

```
purchases_sparse
```

```
<4372x3684 sparse matrix of type '<class 'numpy.int32'>'
        with 267615 stored elements in Compressed Sparse Row format>
```

So, we have a total of 4372 Customers & 3684 items. The sparsity of the matrix is 98.32% as calculated below:

```
matrix_size = purchases_sparse.shape[0]*purchases_sparse.shape[1] # Number of possible interactions in the matrix
num_purchases = len(purchases_sparse.nonzero()[0]) # Number of items interacted with
sparsity = 100*(1 - (num_purchases/matrix_size))
sparsity
```

```
98.33846047247661
```

To proceed with our analysis, we've created a training dataset with some of the user-item pairs set to 0(masked). This would be used in later stages to verify the results of the recommendation.

We then use the training set (sparse matrix of user x item) for implementing ALS for implicit feedback with the below parameters:

> ***training_set*** *- Our matrix of ratings with shape m x n, where m is the number of users and n is the number of items.*
>
> ***lambda_val*** *- Used for regularization during alternating least squares.*
>
> ***alpha*** *- The parameter associated with the confidence matrix.*
>
> ***iterations*** *- The number of times to alternate between both user feature vector and item feature vector in ALS*
>
> ***rank_size*** *- The number of latent features in the user/item feature vectors.*
>
> ***seed*** *- Set the seed for reproducible results*

The function returns a vector for users and items from which we can obtain the rating for each point in the original matrix.

Below figure shows the sample of first 5 items from the total items:

```
In [19]: user_vecs, item_vecs = implicit_weighted_ALS(product_train, lambda_val = 0.1, alpha = 15, iterations = 1,
                                                      rank_size = 20)

In [20]: user_vecs[0,:].dot(item_vecs).toarray()[0,:5]

Out[20]: array([ 1.20009721e-02,  1.74633252e-03,  1.09597250e-03,  1.60241682e-05,
                -2.51953119e-03])
```

To evaluate the performance of our recommendation, we calculate the mean area under the curve (AUC) for users with the items which were masked in the previous steps & also for the most popular items to compare against using the below parameters:

For Users with masked items:

> ***predictions****: the prediction output*
>
> ***test****: the actual target result we are comparing to*

For most popular items:

> ***training_set*** *- The training set where a certain percentage of the original user/item interactions are reset to zero to hide them from the model*
>
> ***predictions*** *- The matrix of our predicted ratings for each user/item pair as output from the implicit MF. These should be stored in a list, with user vectors as item zero and item vectors as item one.*

altered_users *- The indices of the users where at least one user/item pair was altered.*

test_set *- The test set constucted earlier from the above.*

Both the above function returns the calculated AUC

```
In [26]: calc_mean_auc(product_train, product_users_altered,
                       [sparse.csr_matrix(user_vecs), sparse.csr_matrix(item_vecs.T)], product_test)
         # AUC for our recommender system
Out[26]: (0.87, 0.814)
```

As seen from the above results, the AUC for users with the items which were masked is higher (0.87) than the AUC for popular items (0.814)

Next, we find the items which the user has already bought & implement a function to recommend the top items matching to the purchased items using the approach below:

To find items already bought, we use the below parameters:

customer_id *- customer's id number that we want to see prior purchases of at least once*

mf_train *- The initial ratings training set used*

customers_list *- The array of customers used in the ratings matrix*

products_list *- The array of products used in the ratings matrix*

item_lookup *- Unique product ID/product descriptions available*

The above function is used to retrieve the list of items which the user has already purchased

```
In [30]: customers_arr[:5]
Out[30]: array([12346, 12347, 12348, 12349, 12350], dtype=int64)

In [31]: get_items_purchased(12346, product_train, customers_arr, products_arr, item_lookup)
```

Out[31]:

	StockCode	Description
2060	23166	MEDIUM CERAMIC TOP STORAGE JAR

To recommend top matching items, we use the below parameters:

customer_id *- Input the customer's id number that we want to get recommendations for*

mf_train *- The training matrix we used for matrix factorization fitting*

user_vecs *- the user vectors from our fitted matrix factorization*

item_vecs *- the item vectors from our fitted matrix factorization*

customer_list *- an array of the customer's ID numbers that make up the rows of our ratings matrix*

item_list *- an array of the products that make up the columns of our ratings matrix*

item_lookup *- unique product ID/product descriptions available*

num_items *- The number of items you want to recommend in order of best recommendations. Default is 10.*

The above function will return the top recommended items which are never purchased:

```
In [74]: customers_arr[:5]
Out[74]: array([12346, 12347, 12348, 12349, 12350], dtype=int64)

In [75]: get_items_purchased(12346, product_train, customers_arr, products_arr, item_lookup)
```

Out[75]:

	StockCode	Description
61619	23166	MEDIUM CERAMIC TOP STORAGE JAR

The figure below shows sample recommendations for a customer:

```
In [78]: rec_items(12346, product_train, user_vecs, item_vecs, customers_arr, products_arr, item_lookup,
                   num_items = 10)
```

Out[78]:

	StockCode	Description
0	23165	LARGE CERAMIC TOP STORAGE JAR
1	23167	SMALL CERAMIC TOP STORAGE JAR
2	22963	JAM JAR WITH GREEN LID
3	22962	JAM JAR WITH PINK LID
4	23168	CLASSIC SUGAR DISPENSER
5	22978	PANTRY ROLLING PIN
6	84826	ASSTD DESIGN 3D PAPER STICKERS
7	22979	PANTRY WASHING UP BRUSH
8	23295	SET OF 12 MINI LOAF BAKING CASES
9	22982	PANTRY PASTRY BRUSH

Item-Item Similarity:

The rationale behind implementing item similarity recommender is to provide the user with a personalised recommendation based on the similarity with the item that is chosen.

For this recommender, instead of taking the products at an SKU level, we considered using the grouped product category as the item set. In the original ratings matrix created for the first recommender, the sparsity was 98%. By using the grouped product category, we sought to decrease the sparsity of the matrix, since there is a higher chance of the user buying from a category than buying a particular SKU.

Additionally, since the ratings were taken as a measure of the number of purchases, we normalised the matrix by dividing by the total purchases by a user to have the ratings between 0 and 1.

```
ratings_matrix = ratings_matrix.div(ratings_matrix.sum(axis=1), axis=0)
```

In the grouped product category, the sparsity of the matrix in 93.5%.

Model Building:

We first split the data into train and test.

```
Rating_train_df, Rating_test_df = train_test_split(ratings,
                                    stratify=ratings['CustomerID'],
                                    test_size=0.20,
                                    random_state=42)
```

We then train the recommender 'item_similarity_recommender' provided by the graphlab package using the training dataset.

```
#Train Model
item_sim_model = graphlab.item_similarity_recommender.create(rating_train_data, user_id='CustomerID', item_id='ProductID', target='Rating',
                                                             similarity_type='cosine')

#Make Recommendations:
item_sim_recomm = item_sim_model.recommend(users=range(1,6),k=10)
item_sim_recomm.print_rows(num_rows=50)
```

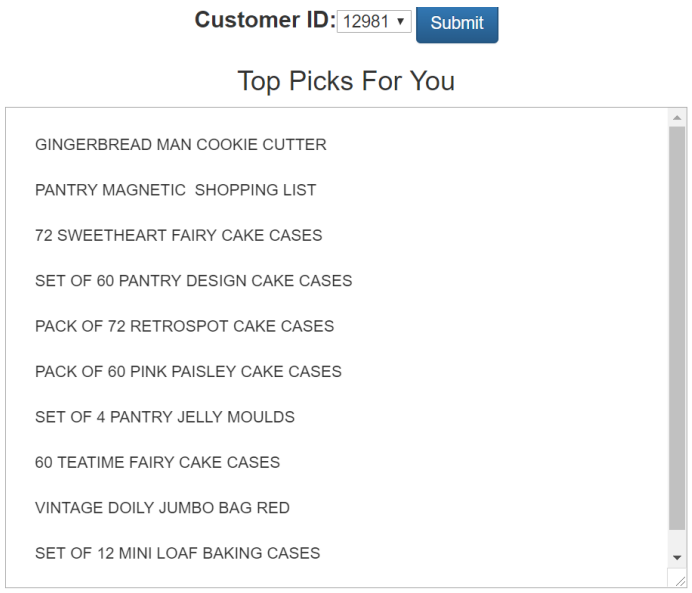
On performing the model evaluation using precision and recall, it is observed that the recall value is very low – 0.13 for a cut-off 10

The low performance of the model can be attributed to the following:

- Matrix sparsity
- Purchasing behaviour as an implicit rating
- Normalising

Since the model evaluation results were not satisfactory, we would not implement this model for our recommendation.

Demo:



The above customer seems to be interested in Baking and Cake accessories.

Analysis Summary & Insights

In summary, by targeting the 3 segments: Clusters 4, 2 & 5 we would be able to capture at least half of our customer base.

Additionally, by catering to just 3 out of the 8 segments we would be able to capture 58% of total sales.

With the strategic objectives obtained from the Association Analysis as well as a clear view on the current customer base (exact customer IDs) we would be able to roll the various campaigns out based on the cluster's purchasing patterns and monitor closely to see how it changes over the next year.

Subsequently this will be a reiterative process of re-clustering, renewing campaigns depending on effectiveness and ultimately drive a higher R, F, M rate which would lead to company growth.

Recommended Strategies

- Mass Market Segment – Lure to buy more by incentivising purchases.
    - "With any purchase of Baking and Cake Accessories get a Set of Doilies Free"
    - "Get 10% off on your next purchase" or "Seasonal Offers – Buy a Christmas Tree and get decorations set at 10% discount"
    - "Get a chance to win an Oven with any purchases in our Baking and Cake Accessories segment"
    - "Limited period offers on Exclusive Festive Products"
- Extravagant Buyers & Champion Segment
    - "Exclusive Access to Preview Sale"
    - Loyalty Points - "Earn more if you spend more"
    - Product Customisation – "Customise a gift for just £2"

Appendix

Data Description

The dataset consists of the following fields:

InvoiceNo:	Nominal, a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation

StockCode	Product (item) code. Nominal, a 5-digit integral number uniquely assigned to each distinct product.
Description:	Product (item) name. Nominal
Quantity	The quantities of each product (item) per transaction. Numeric.
InvoiceDate	Numeric, the day and time when each transaction was generated.
UnitPrice	Numeric, Product price per unit in sterling.
CustomerID:	Customer number. Nominal, a 5-digit integral number uniquely assigned to each customer.
Country	Nominal, the name of the country where each customer resides.

References

Dataset source: https://archive.ics.uci.edu/ml/datasets/online+retail

Clustering: https://stats.stackexchange.com/questions/9016/how-to-define-number-of-clusters-in-k-means-clustering

Segmentation: https://www.putler.com/rfm-analysis/

Recommendation Engine :
http://nbviewer.jupyter.org/github/jmsteinw/Notebooks/blob/master/RecEngine_NB.ipynb

https://www.analyticsvidhya.com/blog/2016/06/quick-guide-build-recommendation-engine-python/

Collaborative Filtering for Implicit Feedback Datasets by Hu, Koren, and Volinsky.